UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/544,274	04/06/2000	Joseph Mulavelil George	AUS000185US1	7599

35525 7590 01/27/2005

IBM CORP (YA)
C/O YEE & ASSOCIATES PC
P.O. BOX 802333
DALLAS, TX 75380

EXAMINER	
LY, ANH	

ART UNIT	PAPER NUMBER
2162	

DATE MAILED: 01/27/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/544,274	GEORGE, JOSEPH MULAVELIL	
	Examiner	Art Unit	
	Anh Ly	2162	

-- The MAILING DATE of this communication appears n the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *16 November 2004*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1,3-12 and 14-48* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1,3-12 and 14-48* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *30 May 2000* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Response to Arguments

1. Applicant's arguments filed 11/16/2004 have been fully considered but they are not persuasive.

Applicant argued that, "none of references teach or suggest referring to a database meta-information class object to determine the structure of a database and then determine a delete action based on the structure of the database determined from the meta-information class object." (Page 12, the 4th paragraph, Page 13, the 1st para. , Page 14, the 2nd para., Page 15, the last para., and Page 18, the last para.)

Ng. Et al. Of 6,385,618 (hereinafter Ng) teaches object relational mapping tools read database schema information and automatically generate a number of class objects whose inter-relationship to the structure of database or schema of the database in the Java environment (col. 2, lines 12-24). Also teaches modifications or update or delete the class object in the object relational database col. 2, lines 35-53). Also Sarkar of 6,418,448 teaches insert, delete and update a transaction of a database by using SQL statement (see fig. 18 and col. 21, lines 7-28).

Applicant argued that, "Nothing in Ng teaches a file that describes the structure and delete actions for tables in a relational database." (Page 17, the 4th paragraph).

Ng teaches the structure of a relational database and with some command for deleting or modification to the class object in that structure (see fig. 7 and col. 2, lines 12-54) and col. 3, lines 60-67 and col. 4, lines 1-10).

Applicant argued that, "there is no teaching or suggestion in Crus to include a delete set null or delete casade operation identifier, for each dependent table of a plurality of table in a relational database.). (Page 19, the 2nd paragraph, Page 24, second para.).

Crus et al. Of 4,947,320 (hereinafter Crus) teaches the commands in SQL such as Load, Insert, Update and Delete commands and their resulting operations. Especially, there are three rules of delete operations: Delete Restrict, Delete Set Null and Delete Casade (col. 5, lines 3-67, col. 17, lines 1-67 and col. 18, lines 1-18) and delete operation is delete the tables in a relational datqabase (col. 24, lines 42-67 and col. 25, lines 1-40).

Applicant argued that, "generating a class object based on a determined structure and determined one or more delete actions, Ng and Elmasri, taken alone or in combination, fail to teach or suggest this feature." (Page 20, second paragraph and Page 25, 2nd para.).

Ng. teaches object relational mapping tools read database schema information and automatically generate a number of class objects whose inter-relationship to the structure of database or schema of the database in the Java environment (col. 2, lines 12-24). Also teaches modifications or update or delete the class object in the object relational database col. 2, lines 35-53) and the structure of a relational database and

with some command for deleting or modification to the class object in that structure (see fig. 7 and col. 2, lines 12-54) and col. 3, lines 60-67 and col. 4, lines 1-10).

2. Claims 1, 3-12 and 14-48 are pending in this application.

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

4. Claims 1, 3-4, 7-12, 14-15 and 17-19 are rejected under 35 U.S.C. 103(a) as being unpatentable over US Patent No. 6,385,618 issued to Ng et al. (hereinafter Ng) in view of Text Book: Fundamentals of Database System (Third Edition) of Ramez Elmasri and Shamkant B. Navathe from Addison-Wisley (hereinafter Elmasri-Navathe) and further in view of in view of US Patent No. 6,418,448 issued to Sarkar.

With respect to claim 1, Ng discloses determining a structure of the relational database (database schema of a relational database: col. 4, lines 23-27 and lines 35-36), wherein determining the structure of the relational database includes referring to a database meta-information class object associated with the relational database (database metadata where information of data concerning data, data definition, characteristics, relationships and external data a database of a database management

system: see abstract, col. 7, lines 60-67 and col. 8, lines 1-18; also see fig. 9); the structure of the relational database as described in the meta-information class object (see fig. 5, the object-relational mapping tool is import database schema, which is containing the information of relationship objects and the class object of the database: col. 6, lines 3-67 and col. 7, lines 12-20).

Ng discloses structure of relational database and schemas of relational database. Ng does not explicitly indicate determining a delete action based on the structure of the relational database and generating database modification commands based on the determined delete action and sending the database modification commands. Elmasri-Navathe discloses active database rules and triggers as referred to as the Event-Condition-Action or ECA-model for the delete operation such as a cascade deletion, the organization or structure of the tables have to be determine to in order to delete tuple that reference the tuple that is being deleted (see rule R4, TOTALSAL4 (page 737 and page 210). In combination, Ng and Elmasri-Navathe do not teach the relational database server in Java via JDBC interface.

However, Sarkar discloses java classes are loaded in the database server (col. 11, lines 45-55; also see col. 6, lines 7-22).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Ng in view of Elmasri-Navathe with the teachings of Sarkar so as to obtain database server of a object relational database locating of elements inside component relational schema with Java classes (col. 6, lines 13-15). This combination would provide a relational database

having database server in the Java classes as argument for the interface of JDBC with SQL in the multi-tier client/server environment (Sarkar – col. 6, lines 20-28) and it is carrying an object SQL query for execution within one or more object relational schema (Sarkar – col. 6, lines 58-65 and querying and viewing multiple object relational schema in the large existing database system (Sarkar – col. 7, lines 10-14) in the deletion of object in the relational database environment.

With respect to claim 3, Ng in view of Elmasri-Navathe discloses a method as discusses in claim 1.

As to the limitation, "wherein the database meta-information class object' encapsulates a dependency structure of the relational database," Ng in view of Elmasri-Navathe does not explicitly indicate that the object classes are encapsulated from the relational database.

However, Sarkar discloses java classes encapsulating the relational data (col. 6, lines 12-20).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Ng in view of Elmasri-Navathe with the teachings of Sarkar so as to obtain database server of a object relational database locating of elements inside component relational schema with Java classes (col. 6, lines 13-15). This combination would provide a relational database having database server in the Java classes as argument for the interface of JDBC with SQL in the multi-tier client/server environment (Sarkar – col. 6, lines 20-28) and it is carrying an object SQL query for execution within one or more object relational schema

(Sarkar – col. 6, lines 58-65 and querying and viewing multiple object relational schema in the large existing database system (Sarkar – col. 7, lines 10-14) in the deletion of object in the relational database environment.

With respect to claims 4 and 7, Ng discloses wherein the database meta-information class object further includes a delete action identifier for each dependent table of a plurality of tables in the relational database and wherein the database meta-information class object is generated based on a file describing the structure and delete actions for tables in the relational database (col. 3, lines 62-67 and col. 7, lines 60-67 and col. 8, lines 1-17; also see fig. 9).

With respect to claim 8, Ng in view of Elmasri-Navathe discloses a method as discusses in claim 1.

As to the limitation, "wherein the file is an Extended Markup Language file, " Ng in view of Elmasri-Navathe does not explicitly indicate that the XML file or document for metadata or constructing arbitrary types by SQL queries.

However, Sarkar discloses Extensible Markup Language (XML) (col. 5, lines 54-60).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Ng in view of Elmasri-Navathe with the teachings of Sarkar so as to obtain database server of a object relational database locating of elements inside component relational schema with Java classes (col. 6, lines 13-15). This combination would provide a relational database having database server in the Java classes as argument for the interface of JDBC with

SQL in the multi-tier client/server environment (Sarkar – col. 6, lines 20-28) and it is carrying an object SQL query for execution within one or more object relational schema (Sarkar – col. 6, lines 58-65 and querying and viewing multiple object relational schema in the large existing database system (Sarkar – col. 7, lines 10-14) in the deletion of object in the relational database environment.

With respect to claims 9-11, Ng discloses wherein the file is further generated based on user input to override default delete action identifiers in the file and wherein the file is further generated based on user input to insert one or more delete constraints in the file for one or more of the tables in the relational database; and commands are SQL statements (col. 3, lines 62-67 and col. 7, lines 60-67 and col. 8, lines 1-17; also see fig. 9; and col.7, lines 16-26).

Claim 12 is essentially the same as claim 1 except that it is directed to a system rather than a method, and is rejected for the same reason as applied to the claim 1 hereinabove.

Claim 14 is essentially the same as claim 3 except that it is directed to a system rather than a method, and is rejected for the same reason as applied to the claim 3 hereinabove.

Claims 15 and 17 are essentially the same as claims 4 and 7 except that they are directed to a system rather than a method, and are rejected for the same reason as applied to the claims 4 and 7 hereinabove.

Claims 18-19 are essentially the same as claims 9-10 except that they are directed to a system rather than a method, and are rejected for the same reason as applied to the claims 9-10 hereinabove.

5. Claims 5-6, 16 and 45 are rejected under 35 U.S.C. 103(a) as being unpatentable over US Patent No. 6,385,618 issued to Ng et al. (hereinafter Ng) in view of Text Book: Fundamentals of Database System (Third Edition) of Ramez Elmasri and Shamkant B. Navathe from Addison-Wisley (hereinafter Elmasri-Navathe) and further in view of US Patent No. 6,418,448 issued to Sarkar and US Patent No. 4,947,320 issued to Crus et al. (hereinafter Crus).

With respect to claims 5-6, Ng in view of Elmasri-Navathe and Sarkar discloses a method of deleting object data from a relational database as discussed in claim 1.

As to limitation, "wherein the delete action identifier is one of cascade delete and nullify columns delete and wherein the delete action is one of cascade delete and nullify columns delete," Ng in view of Elmasri-Navathe and Sarkar does not explicitly indicate that the cascade and nullify column delete based on the relational database schema.

However, Crus discloses delete action identifier such as cascade delete, delete set null as well as nullity columns delete as claimed (col. 5, lines 3-67, col. 6, lines 1-36, col. 16, lines 60-67, col. 17, lines 1-67 and col. 18, lines 1-18).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Ng in view of Elmasri-

Navathe and Sarkar with the teachings of Crus so as to obtain a method of deleting object data from a relational database. This combination would provide an improved method for enforcing referential constraints. The method is useful in any database management system in which records of data are manipulated in response to operations, which may affect multiple records (Crus – col. 3, lines 4-15) in the deletion of object from an object-relational system in a customizable and database independent manner environment.

Claim 16 is essentially the same as claim 5 except that it is directed to a system rather than a method, and is rejected for the same reason as applied to the claim 5 hereinabove.

With respect to claim 45, Ng in view of Elmasri-Navathe and Sarkar discloses a method of deleting object data from a relational database as discussed in claim 43.

As to the limitation, "wherein one or more delete actions is at least one of cascade delete and nullify columns delete," Ng in view of Elmasri-Navathe and Sarkar does not explicitly indicate that the cascade and nullify column delete based on the relational database schema.

However, Crus discloses delete action identifier such as cascade delete, delete set null as well as nullity columns delete as claimed (col. 5, lines 3-67 and col. 6, lines 1-36).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Ng in view of Elmasri-Navathe and Sarkar with the teachings of Crus so as to obtain a method of deleting

object data from a relational database. This combination would provide an improved method for enforcing referential constraints. The method is useful in any database management system in which records of data are manipulated in response to operations, which may affect multiple records (Crus – col. 3, lines 4-15) in the deletion of object from an object-relational system in a customizable and database independent manner environment.

6. Claims 20-21, 24, 25-26, 27-28, 31-34, 35-36, 39-42, 43 and 46-48 are rejected under 35 U.S.C. 103(a) as being unpatentable over US Patent No. 6,385,618 issued to Ng et al. (hereinafter Ng) in view of Text Book: Fundamentals of Database System (Third Edition) of Ramez Elmasri and Shamkant B. Navathe from Addison-Wisley (hereinafter Elmasri-Navathe).

With respect to claim 20, Ng discloses determining a structure of the relational database (col. 8, lines 64-67 and col. 9, lines 11-20; database schema of a relational database: col. 4, lines 23-27 and lines 35-36).

Ng discloses the structure of relational database based on the internal data structure, also known as database data structure, representing the schema of relational database and be enabled change such as add a column (col. 4, lines 32-45), and deletion or manipulation operation. Ng does not explicitly teach determining one or more delete actions based on the structure of the relational database and generating the

class object based on the determined structure and the determined one or more delete actions.

However, Elmasri-Navathe discloses class object from the trigger will be generated based on the determined one or more delete action. If the delete action is a cascade deletion, the organization or structure of the tables have to be determine to in order to delete tuples that reference the tuple that is being deleted (see page 737, rule R4, TOTOLSAL4 and page 210).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Ng with the teachings of Elmasri-Navathe so as to obtain deletion action for the structure of relational database (page 737 and page 210). This combination would provide an improved method for enforcing referential constraints. The method is useful in any database management system in which records of data are manipulated in response to operations, which may affect multiple records the deletion of object from an object-relational system in a customizable and database independent manner environment.

With respect to claims 21 and 24, Ng discloses wherein generating the class object includes encapsulating information identifying the structure of the relational database and the one or more delete actions and discloses wherein the structure of the relational database and the one or more delete actions are determined from a file describing the structure and delete actions for tables in the relational database (col. 3, lines 62-67 and col. 7, lines 60-67 and col. 8, lines 1-17; also see fig. 9; and col.7, lines 16-26).

With respect to claims 25-26, Ng discloses wherein the file is further generated based on user input to override default delete action identifiers in the file and wherein the file is further generated based on user input to insert one or more delete constraints in the file (col. 3, lines 62-67 and col. 7, lines 60-67 and col. 8, lines 1-17; also see fig. 9).

Claim 27 is essentially the same as claim 20 except that it is directed to a system rather than a method, and is rejected for the same reason as applied to the claim 20 hereinabove.

Claim 28 is essentially the same as claim 21 except that it is directed to a system rather than a method, and is rejected for the same reason as applied to the claim 21 hereinabove.

With respect to claim 31, Ng discloses wherein the means for determining the structure of the relational database and the means for determining the one or more delete actions determine the structure and one or more delete actions from a file describing the structure and delete actions of tables in the relational database (col. 3, lines 62-67 and col. 7, lines 60-67 and col. 8, lines 1-17).

With respect to claim 32, Ng discloses wherein the file is generated based on Java Database Connectivity (JDBC) database metadata associated with the relational database (col. 7, lines 60-67).

Claims 33-34 are essentially the same as claims 25-26 except that it is directed to a system rather than a method (col. 3, lines 62-67 and col. 7, lines 60-67 and col. 8,

lines 1-17; also see fig. 9), and is rejected for the same reason as applied to the claims 25-26 hereinabove.

Claim 35 is essentially the same as claim 20 except that it is directed to a computer product rather than a method, and is rejected for the same reason as applied to the claim 20 hereinabove.

Claim 36 is essentially the same as claim 21 except that it is directed to a computer product rather than a method, and is rejected for the same reason as applied to the claim 21 hereinabove.

With respect to claim 39, Ng discloses wherein the means for determining the structure of the relational database and the means for determining the one or more delete actions determine the structure and one or more delete actions from a file describing the structure and delete actions of tables in the relational database (col. 3, lines 62-67 and col. 7, lines 60-67 and col. 8, lines 1-17).

Claim 40 is essentially the same as claim 32 except that it is directed to a computer product rather than an apparatus, and is rejected for the same reason as applied to the claim 32 hereinabove.

Claims 41-42 are essentially the same as claims 25-26 except that it is directed to a computer program product rather than a method, and is rejected for the same reason as applied to the claims 25-26 hereinabove.

With respect to claim 43, Ng discloses a meta-information class for determining a structure of the relational database (col. 8, lines 64-67 and col. 9, lines 11-20; database schema of a relational database: col. 4, lines 23-27 and lines 35-36).

Ng discloses the structure of relational database based on the internal data structure, also known as database data structure, representing the schema of relational database and be enabled change such as add a column (col. 4, lines 32-45), and deletion or manipulation operation. Ng does not explicitly teach one or more delete actions based on the structure of the relational database; and a database meta-information generator class for generating the class object based on the determined structure and determined one or more delete actions.

However, Elmasri-Navathe discloses class object from the trigger will be generated based on the determined one or more delete action. If the delete action is a cascade deletion, the organization or structure of the tables have to be determine to in order to delete tuples that reference the tuple that is being deleted (see page 737, rule R4, TOTOLSAL4 and page 210).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Ng with the teachings of Elmasri-Navathe so as to obtain deletion action for the structure of relational database (page 737 and page 210). This combination would provide an improved method for enforcing referential constraints. The method is useful in any database management system in which records of data are manipulated in response to operations, which may affect multiple records the deletion of object from an object-relational system in a customizable and database independent manner environment.

With respect to claim 46, Ng discloses determining a structure of the relational database (col. 8, lines 64-67 and col. 9, lines 11-20; database schema of a relational database: col. 4, lines 23-27 and lines 35-36).

Ng discloses the structure of relational database based on the internal data structure, also known as database data structure, representing the schema of relational database and be enabled change such as add a column (col. 4, lines 32-45), and deletion or manipulation operation. Ng does not explicitly teach determining one or more default delete actions based on the structure of the relational database; receiving user input to modify the one or more default delete actions; and generating the class object based on the determined structure, the determined one or more delete actions and the user input.

However, Elmasri-Navathe discloses class object from the trigger will be generated based on the determined one or more delete action. If the delete action is a cascade deletion, the organization or structure of the tables have to be determine to in order to delete tuples that reference the tuple that is being deleted (see page 737, rule R4, TOTOLSAL4 and page 210).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Ng with the teachings of Elmasri-Navathe so as to obtain deletion action for the structure of relational database (page 737 and page 210). This combination would provide an improved method for enforcing referential constraints. The method is useful in any database management system in which records of data are manipulated in response to operations, which may

affect multiple records the deletion of object from an object-relational system in a customizable and database independent manner environment.

With respect to claims 47-48, Ng discloses wherein the file is further generated based on user input to override default delete action identifiers in the file and wherein the file is further generated based on user input to insert one or more delete constraints in the file for one or more of the tables in the relational database (col. 4, lines 45-67, col. 6, lines 42-64 and col. 7, lines 9-67).

7. Claims 22-23, 29-30 and 37-38 are rejected under 35 U.S.C. 103(a) as being unpatentable over US Patent No. 6,385,618 issued to Ng et al. (hereinafter Ng) in view of Text Book: Fundamentals of Database System (Third Edition) of Ramez Elmasri and Shamkant B. Navathe from Addison-Wisley (hereinafter Elmasri-Navathe) and further in view of US Patent No. 4,947,320 issued to Crus et al. (hereinafter Crus).

With respect to claims 22-23, Ng in view of Elmasri-Navathe discloses a method of deleting object data from a relational database as discussed in claim 20.

As to the limitation, "wherein the delete action identifier is one of cascade delete and nullify columns delete and wherein the delete action is one of cascade delete and nullify columns delete," Ng in view of Elmasri-Navathe does not explicitly indicate that the cascade and nullify column delete based on the relational database schema.

However, Crus discloses delete action identifier such as cascade delete, delete set null as well as nullity columns delete as claimed (col. 5, lines 3-67, col. 6, lines 1-36, col. 16, lines 60-67, col. 17, lines 1-67 and col. 18, lines 1-18).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Ng in view of Elmasri-Navathe with the teachings of Crus so as to obtain a method of deleting object data from a relational database. This combination would provide an improved method for enforcing referential constraints. The method is useful in any database management system in which records of data are manipulated in response to operations, which may affect multiple records (Crus – col. 3, lines 4-15) in the deletion of object from an object-relational system in a customizable and database independent manner environment.

Claims 29-30 are essentially the same as claims 22-23 except that it is directed to a system rather than a method, and is rejected for the same reason as applied to the claims 22-23 hereinabove.

Claims 37-38 are essentially the same as claims 22-23 except that it is directed to a computer program product rather than a method, and is rejected for the same reason as applied to the claims 22-23 hereinabove.

8. Claim 44 is rejected under 35 U.S.C. 103(a) as being unpatentable over US Patent No. 6,385,618 issued to Ng et al. (hereinafter Ng) in view of Text Book: Fundamentals of Database System (Third Edition) of Ramez Elmasri and Shamkant B. Navathe from Addison-Wisley (hereinafter Elmasri-Navathe) and further in view of US Patent No. 6,418,448 issued to Sarkar.

With respect to claim 44, Ng in view of Elmasri-Navathe discloses a program product as discusses in claim 43.

As to the limitation, "wherein the database meta-information class object encapsulates a dependency structure of the relational database," Ng in view of Elmasri-Navathe does not explicitly indicate that the object classes are encapsulated from the relational database.

However, Sarkar discloses java classes encapsulating the relational data (col. 6, lines 12-20).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Ng in view of Elmasri-Navathe with the teachings of Sarkar so as to obtain database server of a object relational database locating of elements inside component relational schema with Java classes (col. 6, lines 13-15). This combination would provide a relational database having database server in the Java classes as argument for the interface of JDBC with SQL in the multi-tier client/server environment (Sarkar – col. 6, lines 20-28) and it is carrying an object SQL query for execution within one or more object relational schema (Sarkar – col. 6, lines 58-65 and querying and viewing multiple object relational schema

in the large existing database system (Sarkar – col. 7, lines 10-14) in the deletion of object in the relational database environment.

Conclusion

9. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

JEAN M. CORRIELUS
PRIMARY EXAMINER

Contact Information

10. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Anh Ly whose telephone number is (571) 272-4039 or via E-Mail: ANH.LY@USPTO.GOV **or fax to (571) 273-4039**. The examiner can normally be reached on TUESDAY – THURSDAY from 8:30 AM – 3:30 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John Breene, can be reached on (571) 272-4107 or Primary Examiner Jean Corrielus (571) 272-4032.

Any response to this action should be mailed to:

Commissioner of Patents and Trademarks

Washington, D.C. 20231

or faxed to: Central Fax Center (703) 872-9306

ANH LY

JAN. 21st, 2005